Preserving User Privacy in Shared Content

Panagiotis Ilia

Institute of Computer Science
Foundation for Research and Technology – Hellas (FORTH)

<pilia@ics.forth.gr>

Online Social Networks

Facebook, Google+, MySpace, Flickr, Twitter, Tumblr ...

- **Facebook**: More than **1.2b** users currently

  More than **350m** photos uploaded daily.

- **Google+**: **500m** registered users in May 2013 (launched in 2011).

  235m active users per month.

In **OSNs** users create their digital profiles:

- **Connect/communicate with others**
- **Generate and publish their content**

Concerns about user privacy

- Average users don’t care about their privacy
- Access control mechanisms are complicated
- Users are not aware about the implications of their actions.
- Users are unaware about the **”true visibility”** of the **uploaded content**

How to minimize the leakage of Private Information

- ➢ Users must be able to choose what to share to whom
  - ○ Define an effective access control policy
  - ○ Configure SN profile to enforce this policy

Current OSN design:

- ➢ The content publisher is also the content owner.
- ➢ Users can control only self-disclosed information.
  - ❑ Users **cannot** control **shared content** published by others.

Uploaded photos – privacy of the depicted users

- ➢ The photo uploader is considered as the owner of the photo
  - o The uploader is granted full rights on the photo.

- ➢ The depicted users are not considered as co-owners.
  - o They are **not** granted any rights on the photo.
  - o They cannot restrict or removal the photo.

But:

The **tagged users** can **affect the visibility** of the photo

By assigning **permissive privacy settings**

Conflict of interests

- ➢ The will of the uploader goes against the will of the depicted users.
- ➢ The privacy settings of a user are overridden by those of other users.

Scenario: The Sober Tagger

- Alice uploads an *"embarrassing"* photo of co-worker Bob.
- Bob request photo removal – Alice does not remove it.

Scenario: The Silent Tagger

- Alice does not tag Bob, thus Bob is never notified about the photo.

Scenario: The Group Picture

- Bob set the photo as **"private"** – a depicted friend set it as **"public"**

Privacy risk of depicted users

- **User privacy risk for specific item** : $PR(t, i) = \beta_t \times Vis(i, t)$

- **Overall user privacy risk:** $PR(i) = \sum_{t=1}^{l} \beta_t \times Vis(i, t)$

Depends on **sensitivity** and **visibility**

Extend previous risk models - calculating the risk posed by shared content

- ***Intentional risk*** (permissions set by the user)
- ***Unintentional risk*** (permissions set by others)

Privacy risk of presented users - intentional/unintentional risk

2-hop nodes for the “tagged users”

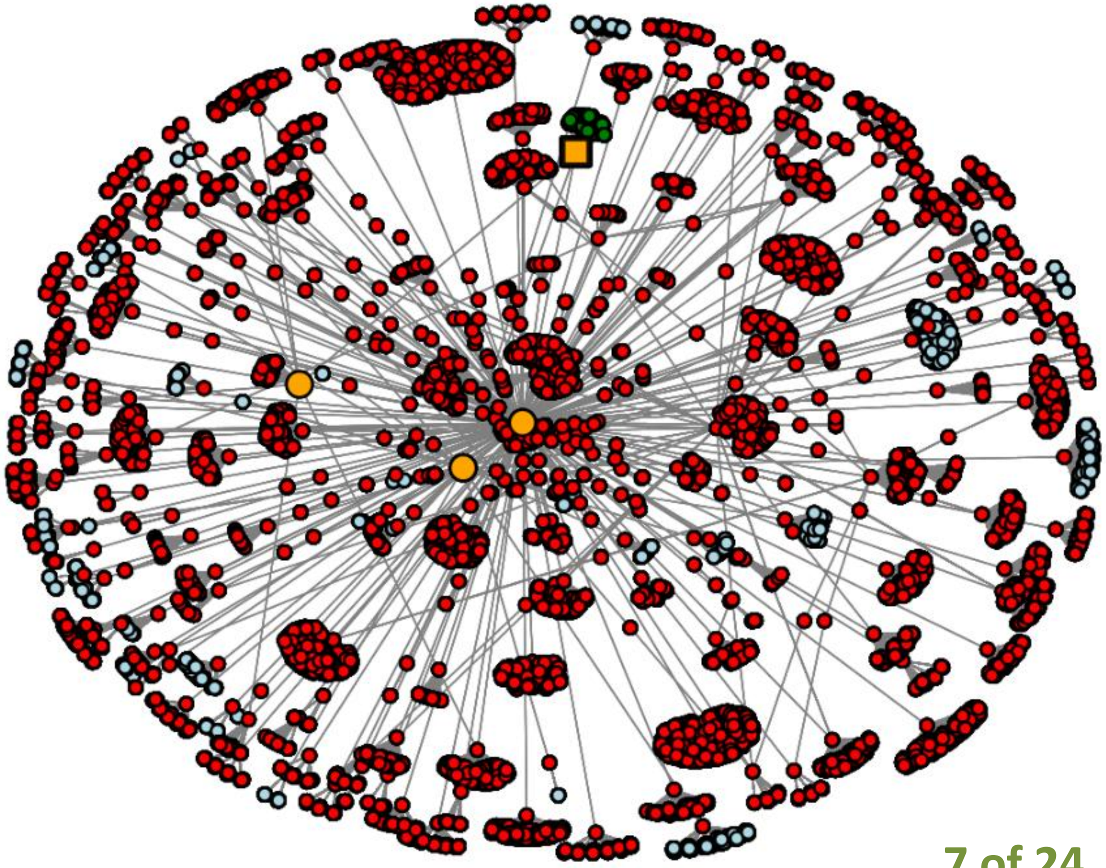
🟧 Point of interest

🟠 Tagged in the photo

🟢 Given access by user

🔴 Given access by others

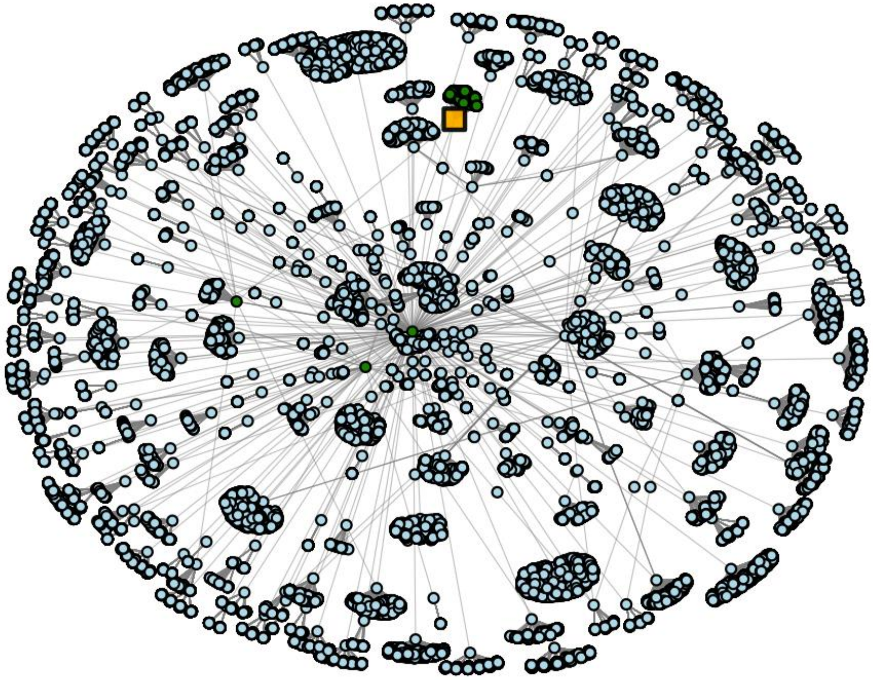
○ Not given access

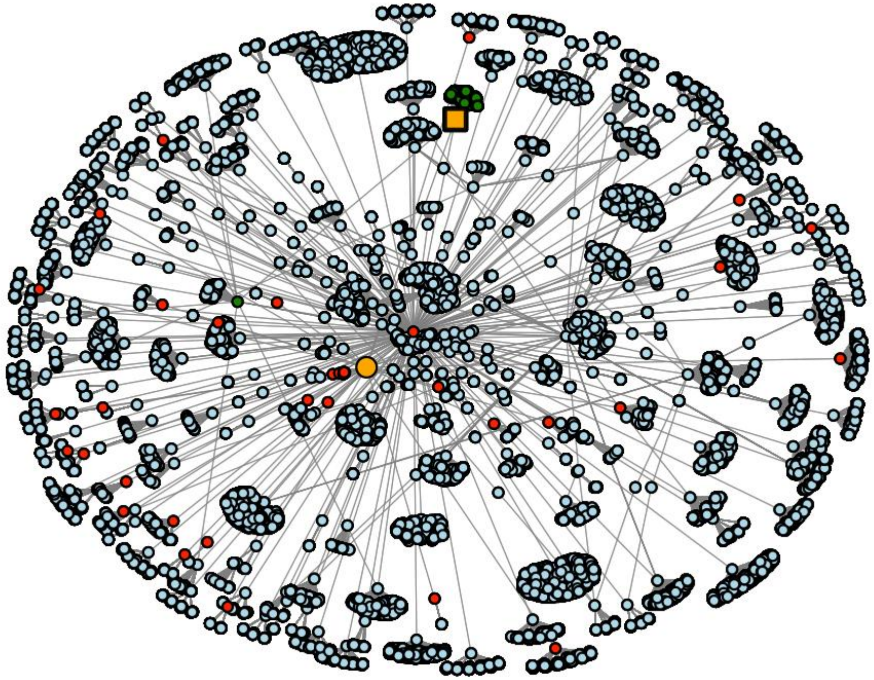


Progression of Privacy risk - intentional/unintentional risk

User of Interest is tagged (UoI)
Friends of UoI gain access
(339 green nodes)

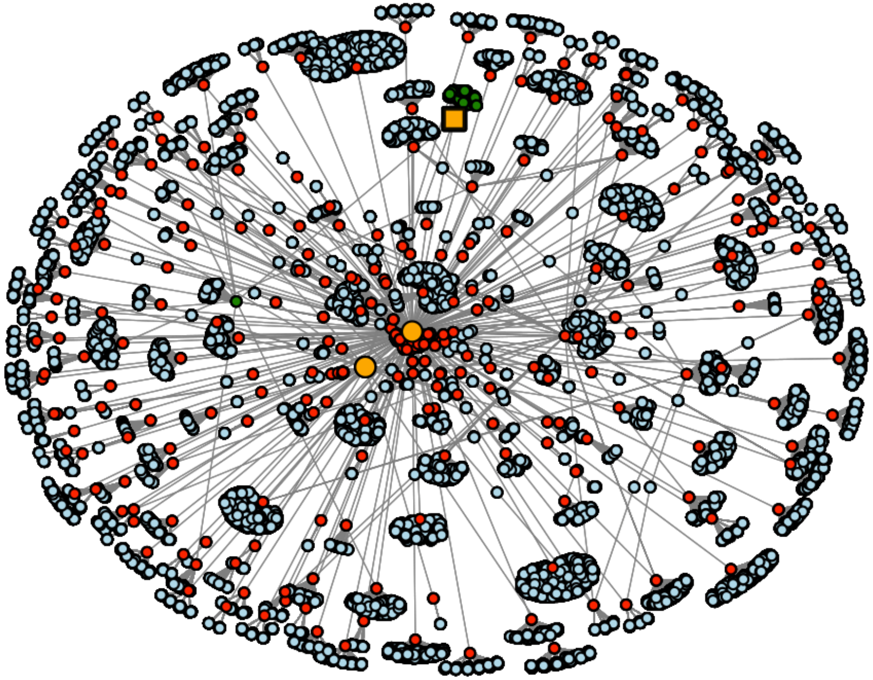
2nd User is tagged
Friends of 2nd User gain access
(51 red nodes)



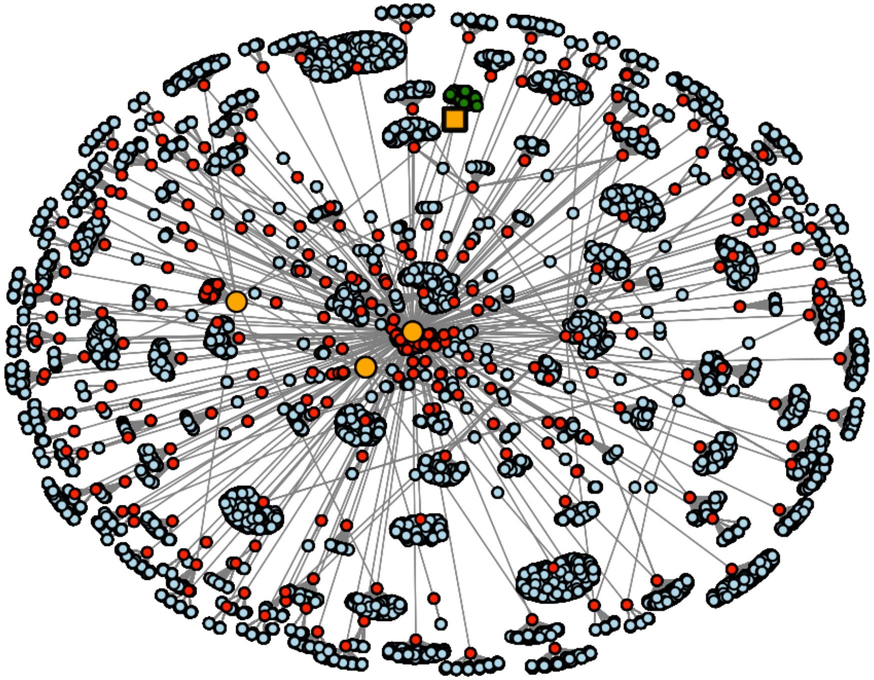


Progression of Privacy risk - intentional/unintentional risk

3nd User is tagged
(379 red nodes)



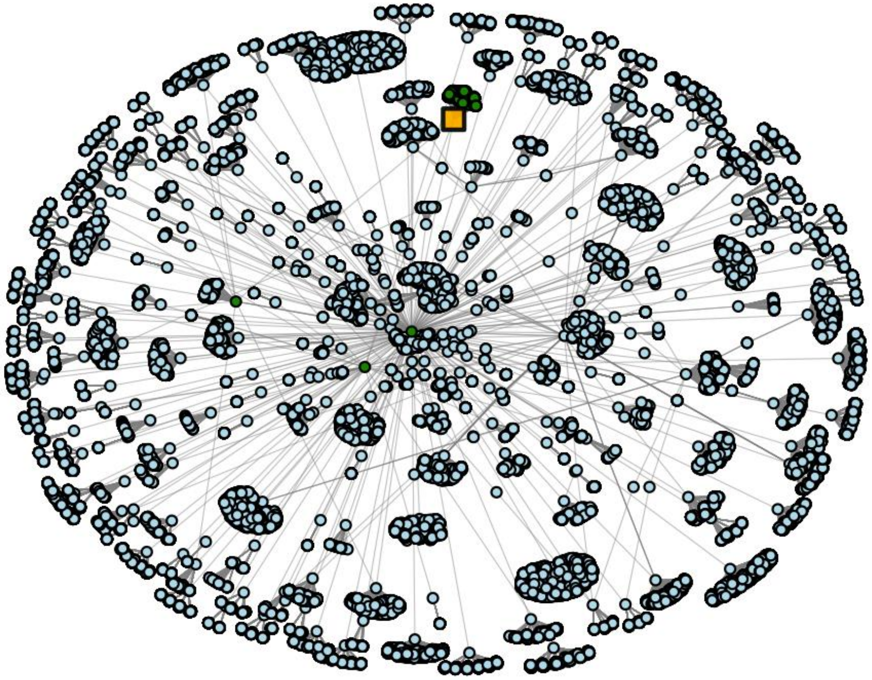
4nd User is tagged
(528 red nodes)



Progression of Privacy risk - intentional/unintentional risk

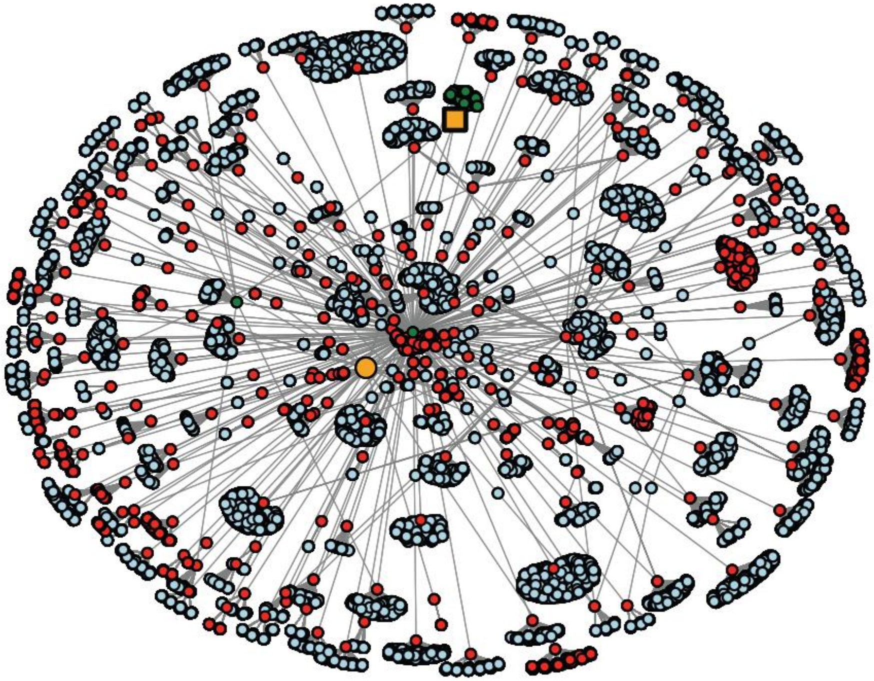
Friends-of-Friends scenario

User of Interest – only Friends



2nd User is tagged

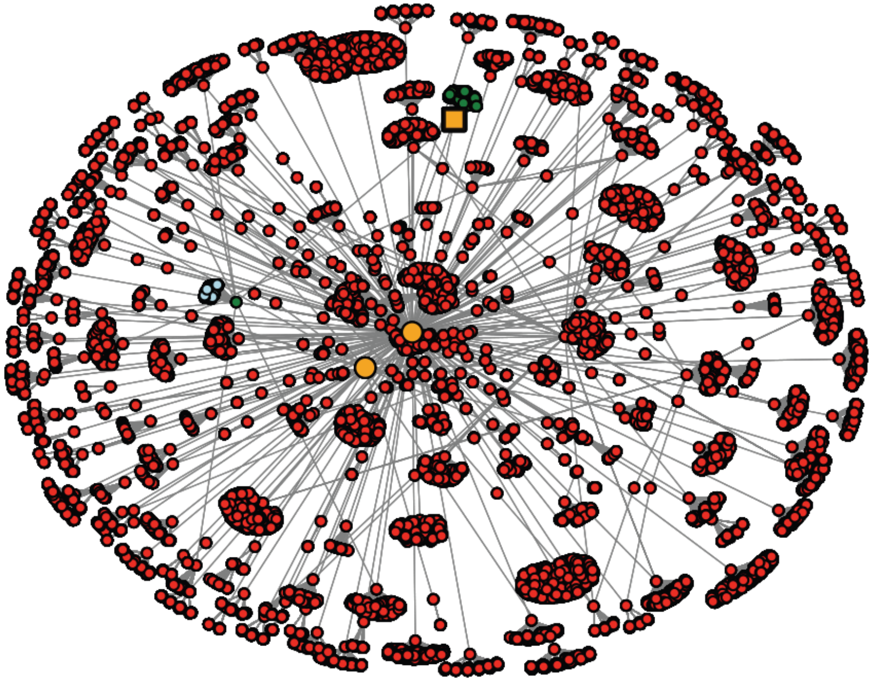
(7.3k red nodes)



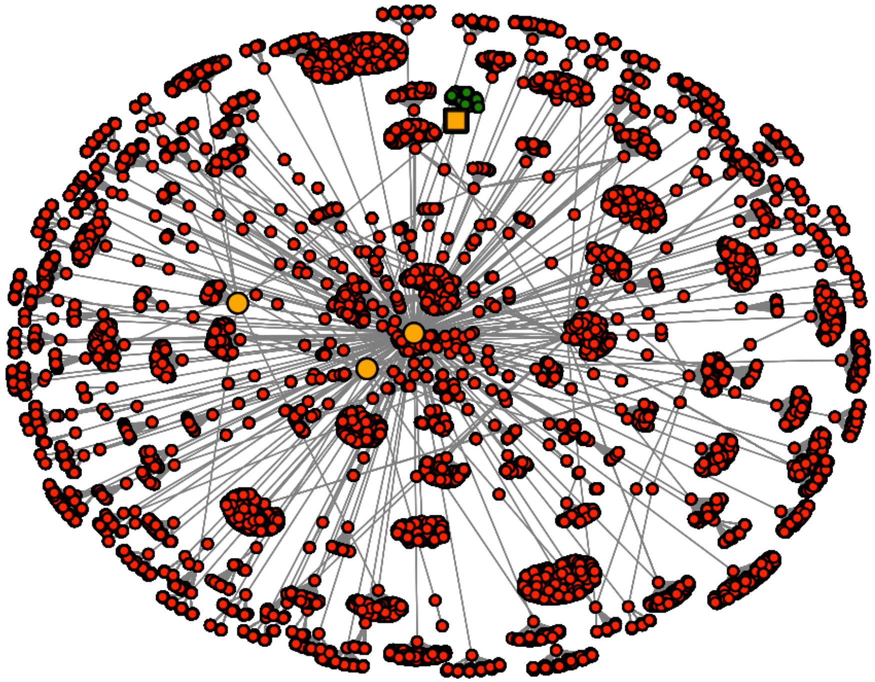
Progression of Privacy risk - intentional/unintentional risk

Friends-of-Friends scenario

3nd User is tagged
(54.5k red nodes)



4nd User is tagged
(54.8k red nodes)



Contributions of this work

- Extend previous risk models - "*intentional*" and "*unintentional*" risk.
  - Takes into account the **access control permissions** of all relevant parties.
  - Takes into account the **position of the parties** within the social graph.

- Design a new fine-grained access control mechanism.
  - Enforce **face-level** access control (according to user's access-list).
  - Handles effectively the **conflicting visibility settings** of the users.
  - Can **inter-operate** with the existing access control mechanisms.

- Proof-of-concept application.
  - Feasibility and applicability of the approach within the OSN infrastructure.

Previous work

○ Survey on user behaviour (why tag/un-tag) , ownership, privacy. **[Besmer, SOUPS 08]**

○ A “negotiation” mechanism. Out-of-band request to the uploader to hide the photo.

○ Does not solve conflict of interests. Follows an allow/deny logic. **[Besmer, SIGCHI 10]**

Rule-based access control

○ Users annotate photos with custom descriptive tags. AC rules according to these tags.

○ Access control on photo-level (allow/deny). **[Klemperer, SIGCHI 12]**

Rule-based mechanism / similar to recommendation systems

○ AC policy according to rules. Classifies new photos and predicts an acceptable rule.

[Squicciarini, HT’ 11]

Security rules for content-based access control

○ Uses the SWRL language. The owner sets complex Positive and Negative rules.

○ Mechanism for resolving conflicting rules. Depends on the owner to set attributes /rules

[Al Bouna, SITIS 12]

Access control mechanisms

- The photo is considered as personally identifiable information (PII).
- "***Allow/Deny***" access control mechanism (photo-level).

However

- Each user's face is also PII (for the particular user).
- Our mechanism **switches the granularity** of the access control …

  … from the level of a **photo** to that of users' **faces**.
- User's privacy settings are enforced upon their face.
- Restrictive user's privacy settings are not overridden by others.

Proposed access control model

Does not affect to photo-level access control

Works on top of the current mechanisms

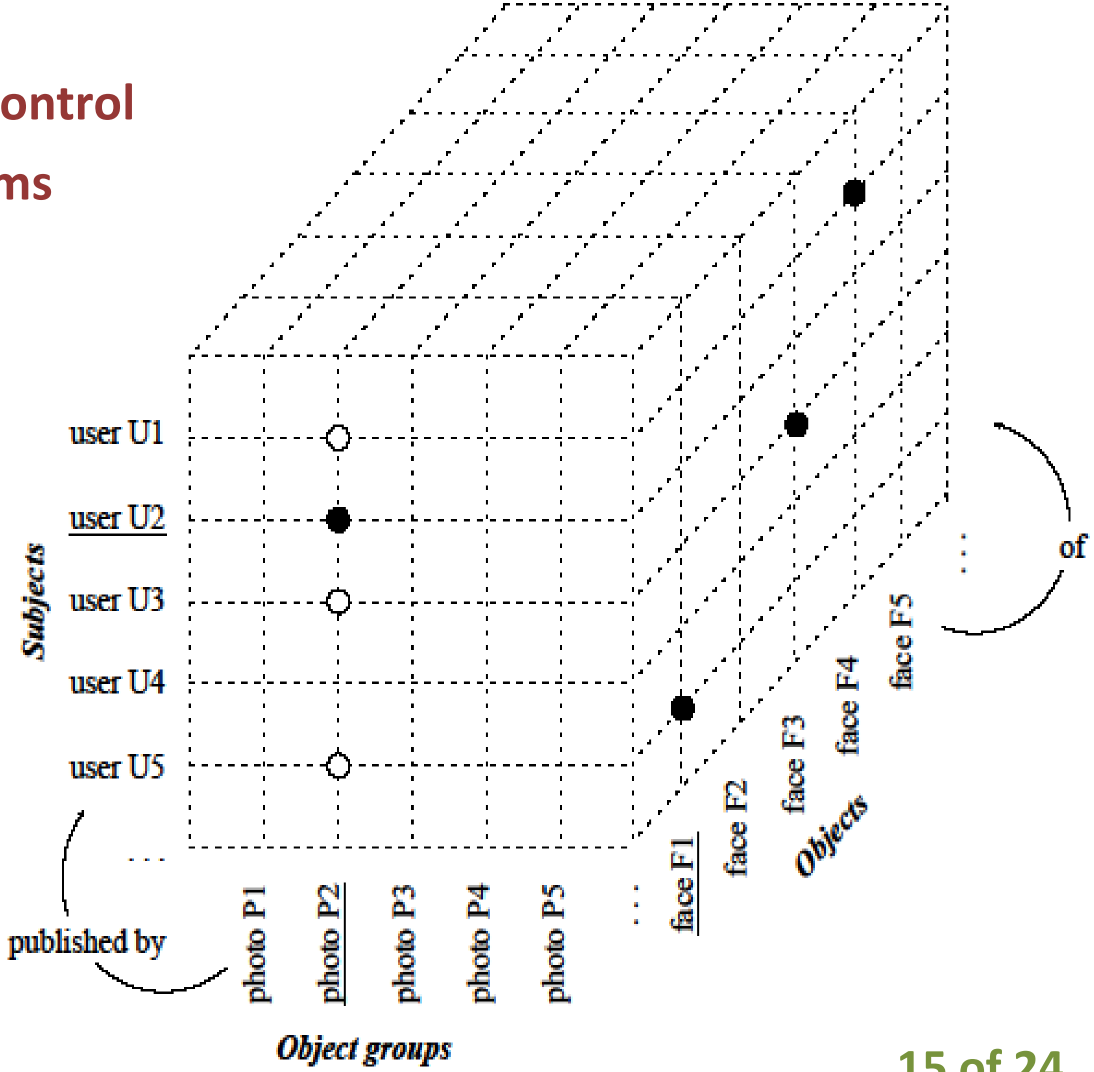
Subjects = Users

Objects = Faces of Users

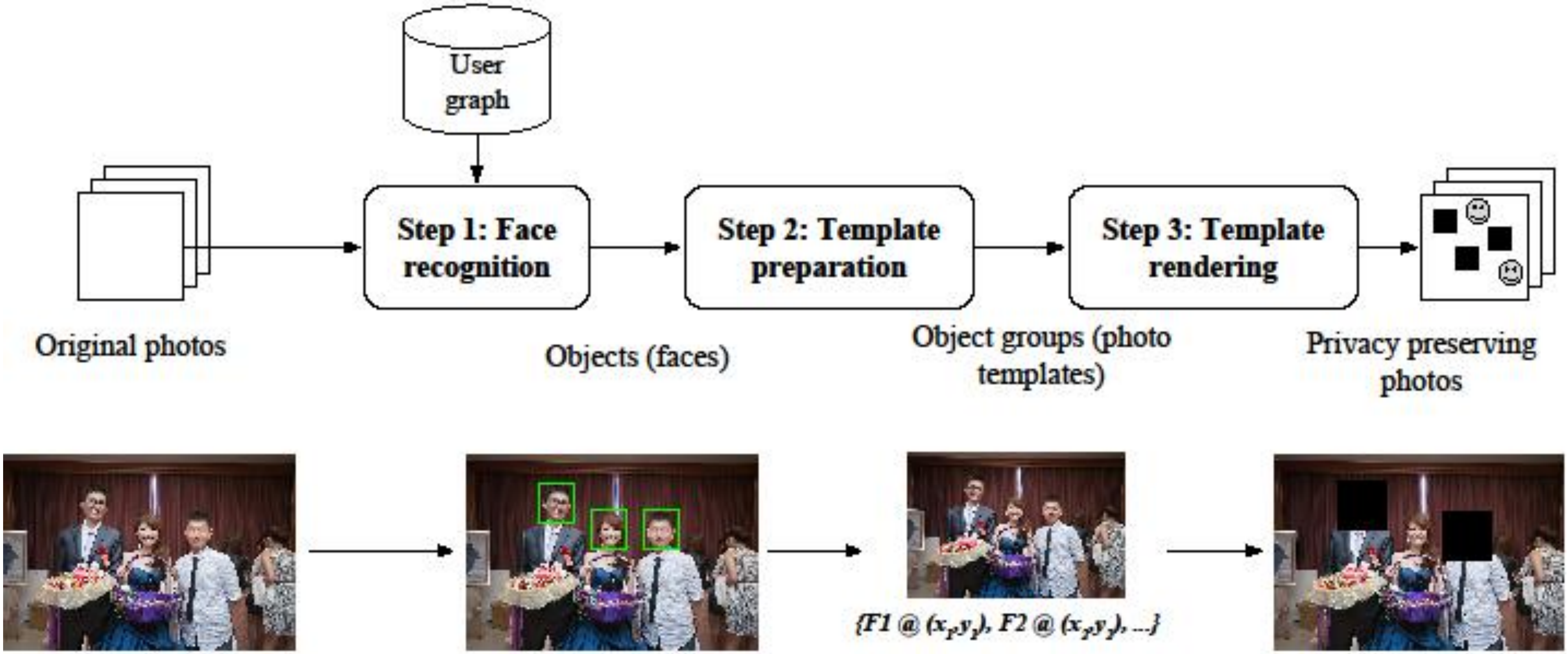
Photo = Group of Objects

Extension of the current OSN access control mechanisms

All the permissions bits are enabled



Overview of the access control approach


User graph
Step 1: Face recognition
Step 2: Template preparation
Step 3: Template rendering
Original photos
Objects (faces)
Object groups (photo templates)
Privacy preserving photos
{F1 @ (x_1,y_1), F2 @ (x_2,y_2), ...}

Overview of the access control design

Step 1: Face Recognition

- When a photo is uploaded, detect the faces and recognize known users.
- Each face becomes an object in the access control model.

Step 2: Template Preparation

- Auto-tagging the identified faces, or tag-suggestion (for verification).
- The users are automatically notified to verify the face validity.
- Tagged users set their face-level access control (access list).

- A small photo (layer) is derived, containing a single hidden face.
- The template is consisted of the original photo and the created layers.

Overview of the access control approach

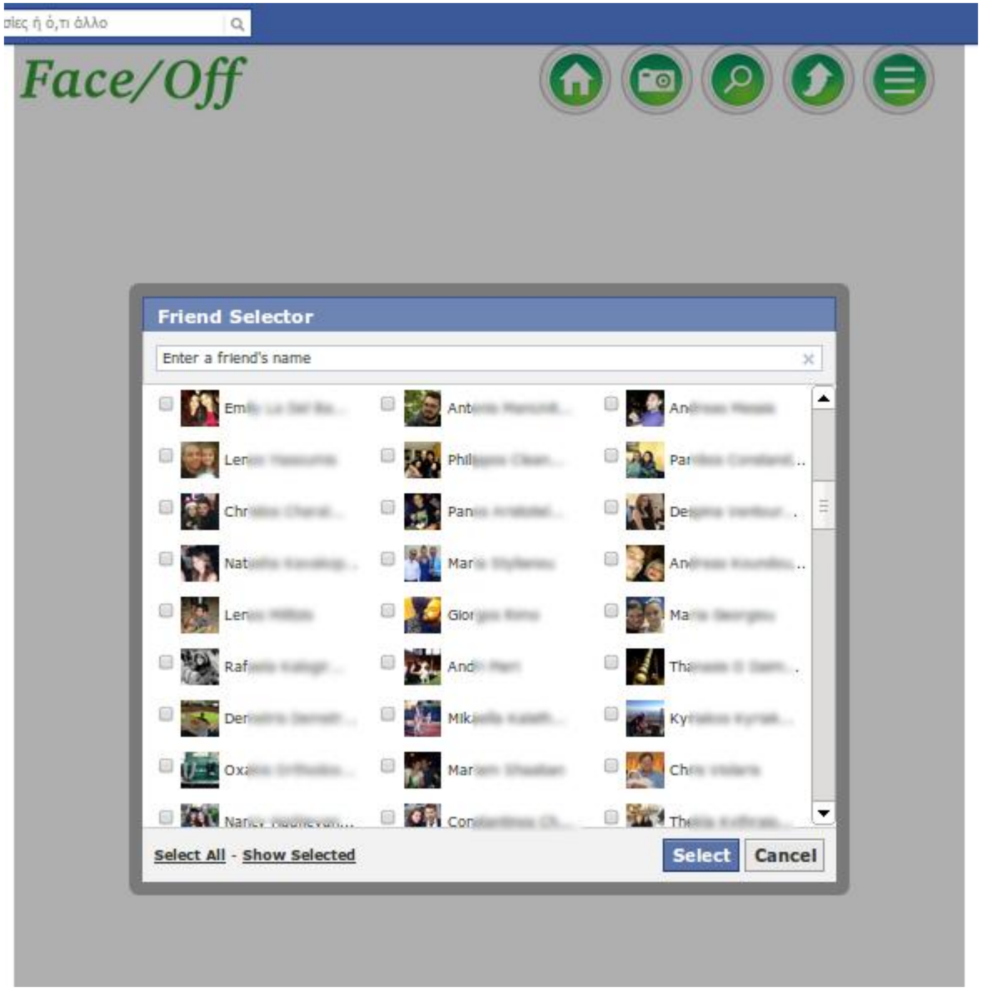
Step 3: Template Rendering

- Determine in constant time the hidden faces (access control matrix)
- The photo is rendered selectively according to who is viewing it.
- The requested photo is created "**on the fly**".
- Superimposing the required layers, on top of the original photo.

User Lists

- The users have a set of personalized friend-lists.
- Every list represent a group of friends with common characteristics.
- These lists are used as access control lists (ACL) for published content.
- A list is created or deleted at any time - users added/removed dynamically.

Preserving user privacy in shared content



σίες ή ό,τι άλλο

Face/Off

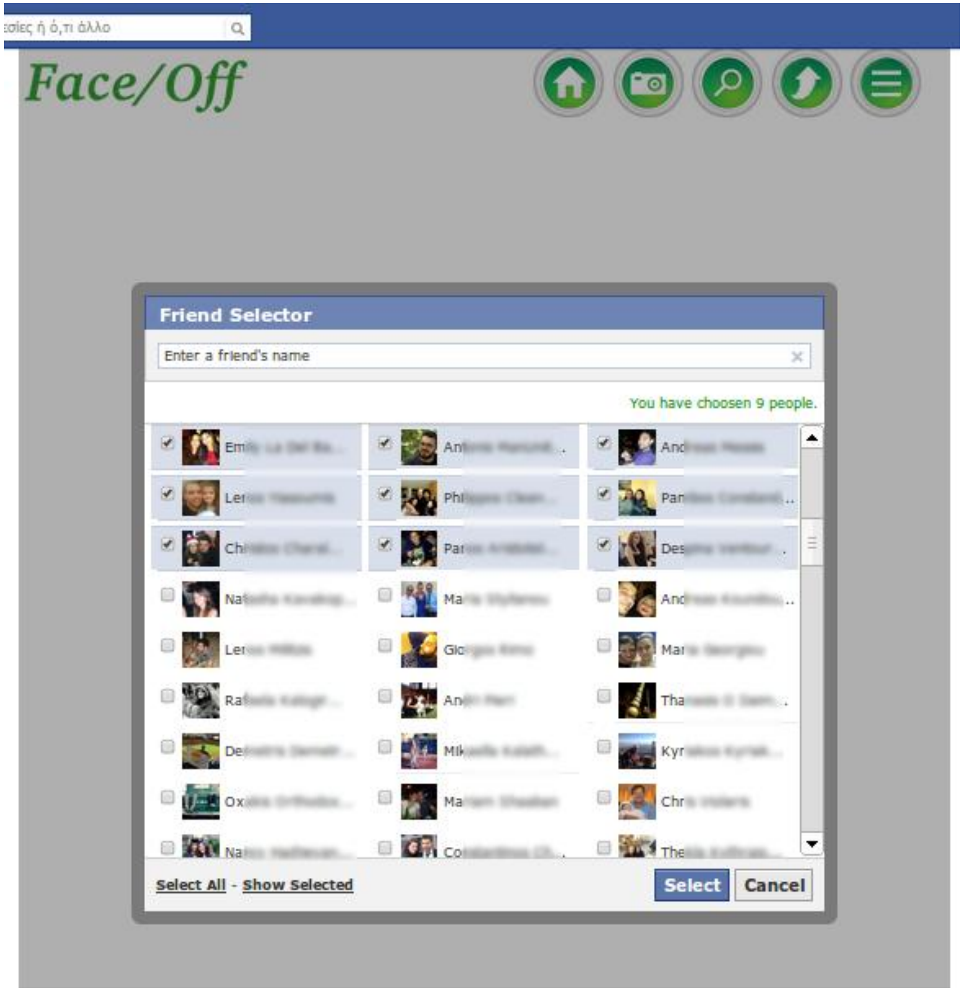
Friend Selector

Enter a friend's name

☐ Em[redacted] ☐ Ant[redacted] ☐ An[redacted]
☐ Ler[redacted] ☐ Phil[redacted] ☐ Par[redacted]
☐ Chr[redacted] ☐ Pan[redacted] ☐ De[redacted]
☐ Nat[redacted] ☐ Mar[redacted] ☐ An[redacted]
☐ Ler[redacted] ☐ Gior[redacted] ☐ Ma[redacted]
☐ Raf[redacted] ☐ And[redacted] ☐ Th[redacted]
☐ Der[redacted] ☐ Mik[redacted] ☐ Ky[redacted]
☐ Ox[redacted] ☐ Mar[redacted] ☐ Ch[redacted]
☐ Nanc[redacted] ☐ Con[redacted] ☐ Th[redacted]

Select All - Show Selected

Select Cancel



σίες ή ό,τι άλλο

Face/Off

Friend Selector

Enter a friend's name

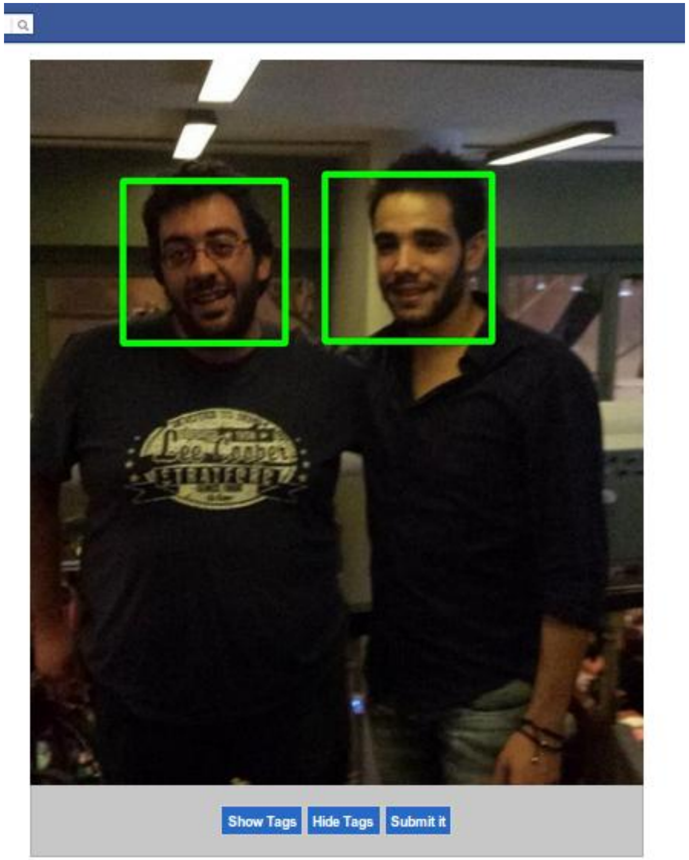
You have choosen 9 people.

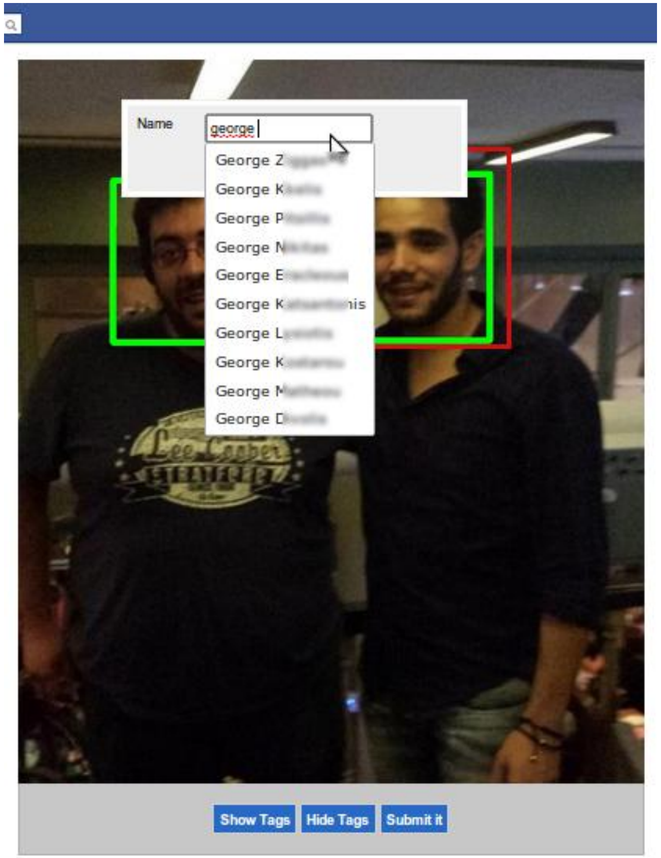
☑ Em[redacted] ☑ An[redacted] ☑ And[redacted]
☑ Ler[redacted] ☑ Phi[redacted] ☑ Par[redacted]
☑ Ch[redacted] ☑ Par[redacted] ☑ Des[redacted]
☐ Na[redacted] ☐ Ma[redacted] ☐ And[redacted]
☐ Ler[redacted] ☐ Gio[redacted] ☐ Mar[redacted]
☐ Ra[redacted] ☐ An[redacted] ☐ Tha[redacted]
☐ De[redacted] ☐ Mi[redacted] ☐ Kyr[redacted]
☐ Ox[redacted] ☐ Ma[redacted] ☐ Chr[redacted]
☐ Na[redacted] ☐ Co[redacted] ☐ The[redacted]

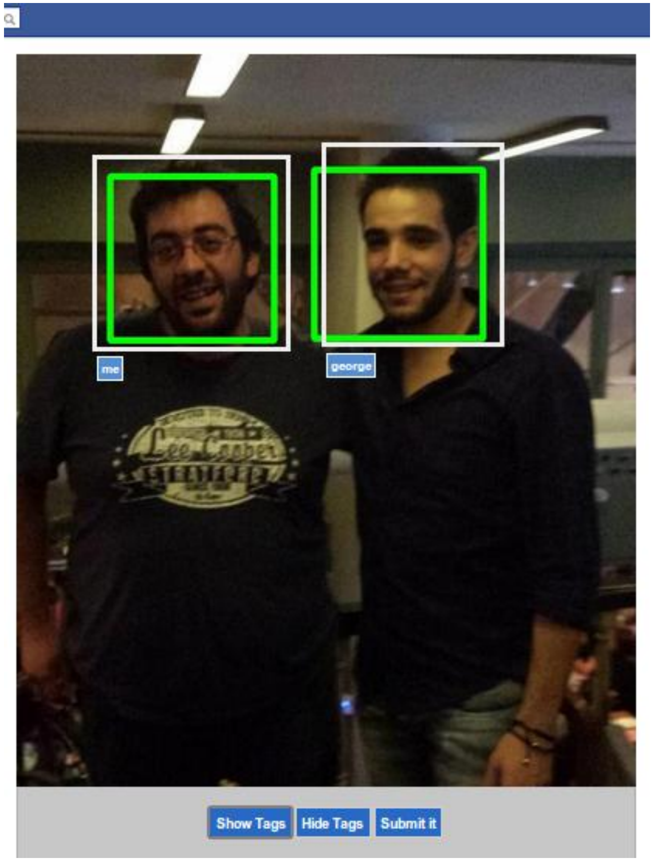
Select All - Show Selected

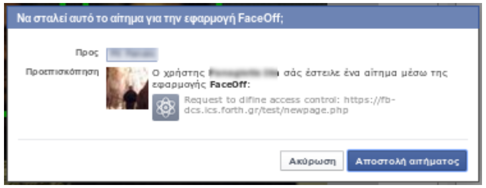
Select Cancel

Photo upload - face detection - tagging - notification









Να σταλεί αυτό το αίτημα για την εφαρμογή FaceOff;

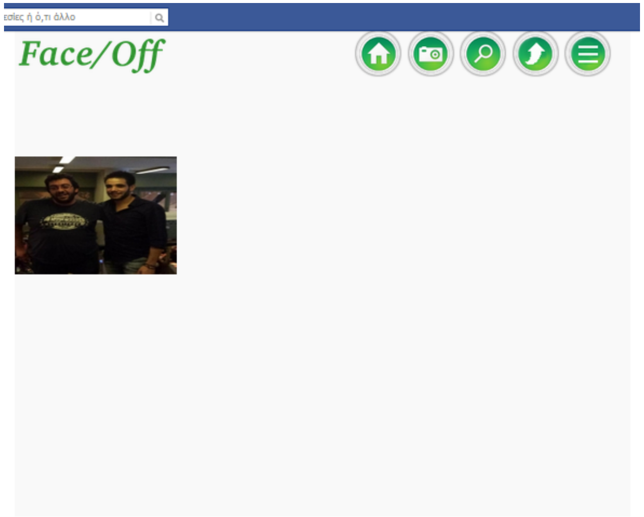
Προς [redacted]

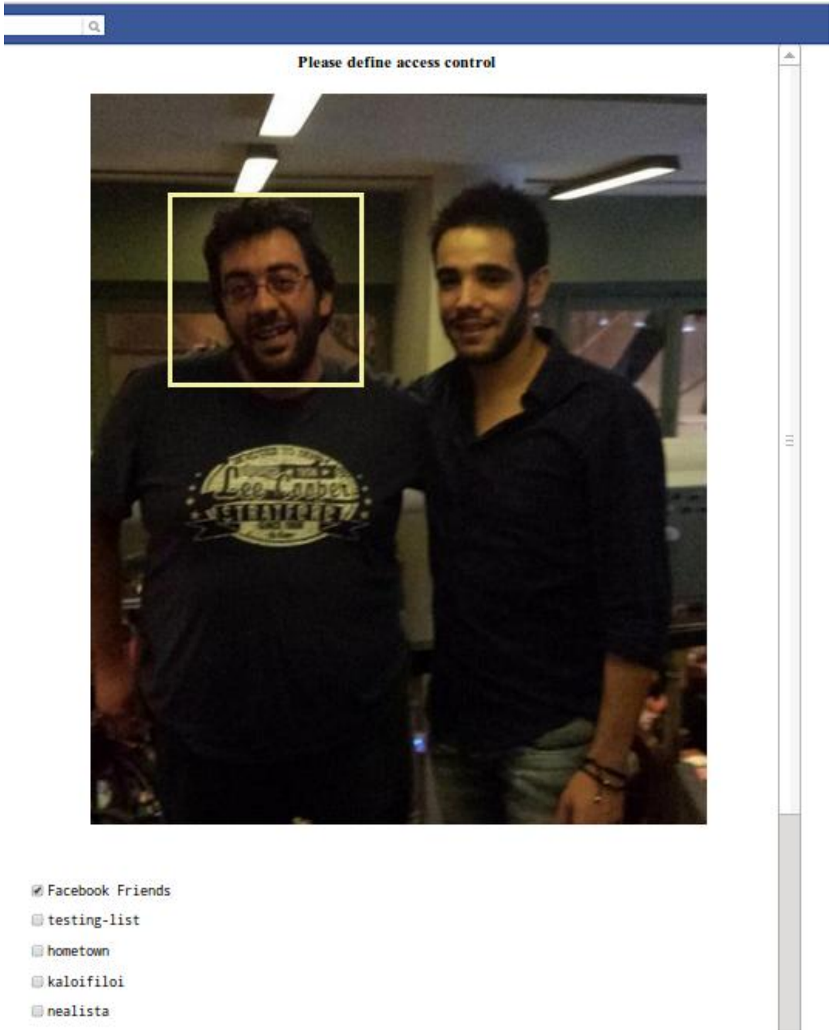
Προεπισκόπηση Ο χρήστης [redacted] σάς έστειλε ένα αίτημα μέσω της εφαρμογής **FaceOff**:

Request to difine access control: https://fb-dcs.ics.forth.gr/test/newpage.php

New photos as thumbnails

Defining Access Lists





Please define access control

☑ Facebook Friends

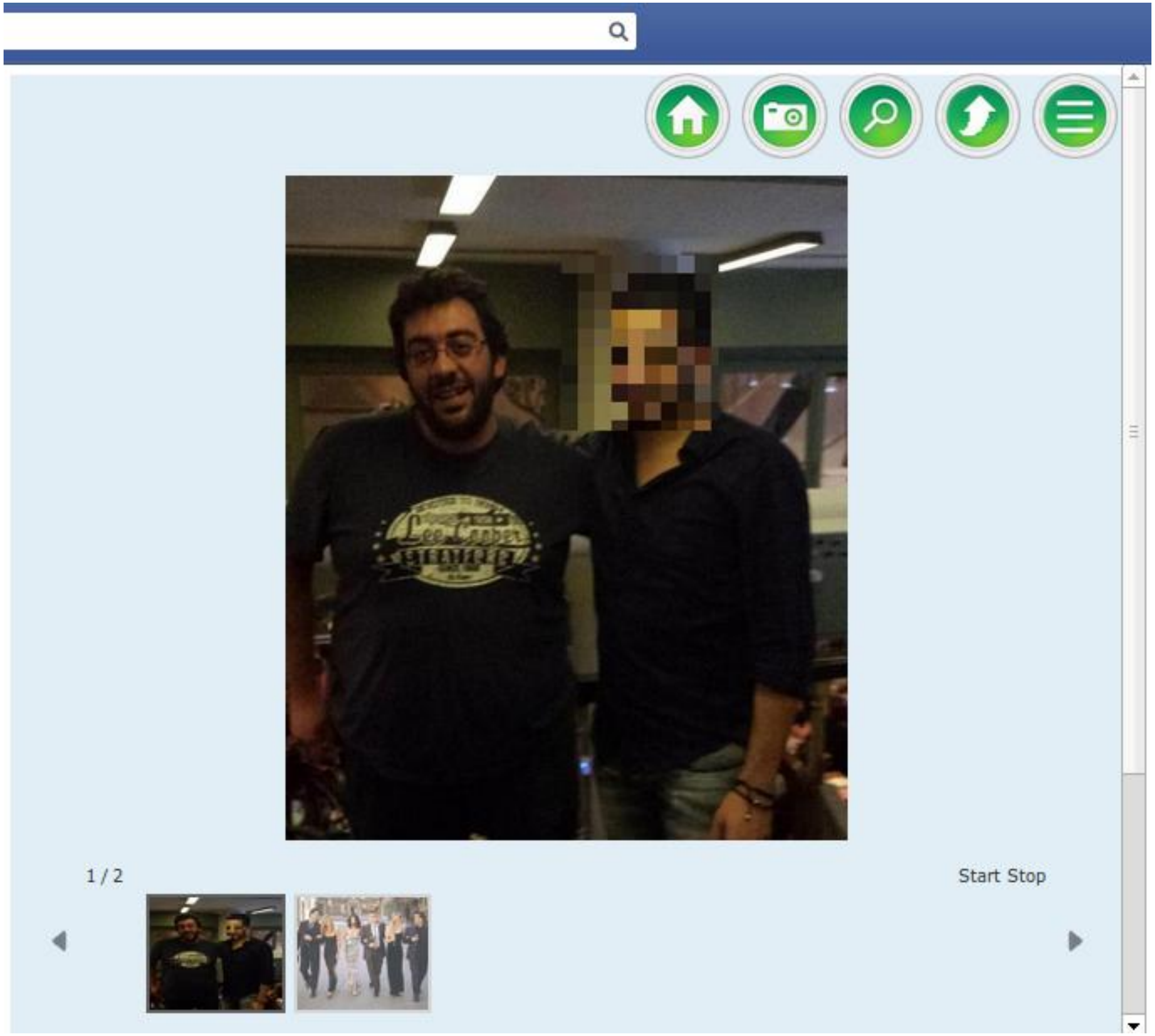
□ testing-list

□ hometown

□ kaloifiloi

□ nealista

Preserving user privacy in shared content



What is next…

Study "*conflict of interests*" in a Decentralized Setting

Is our model feasible for DOSNs?

- Can face identification performed decentralized? (privacy issues?)
- No central authority .. How to enforce the model?
- Permanently modified photos? Or processed "**on the fly**".

Summary

- **Tagged users affect the visibility** of photos – set **permissive privacy settings.**

Conflict of interests

- The will of a user goes against the will of the other depicted users.

  Intentional risk and ***Unintentional risk***

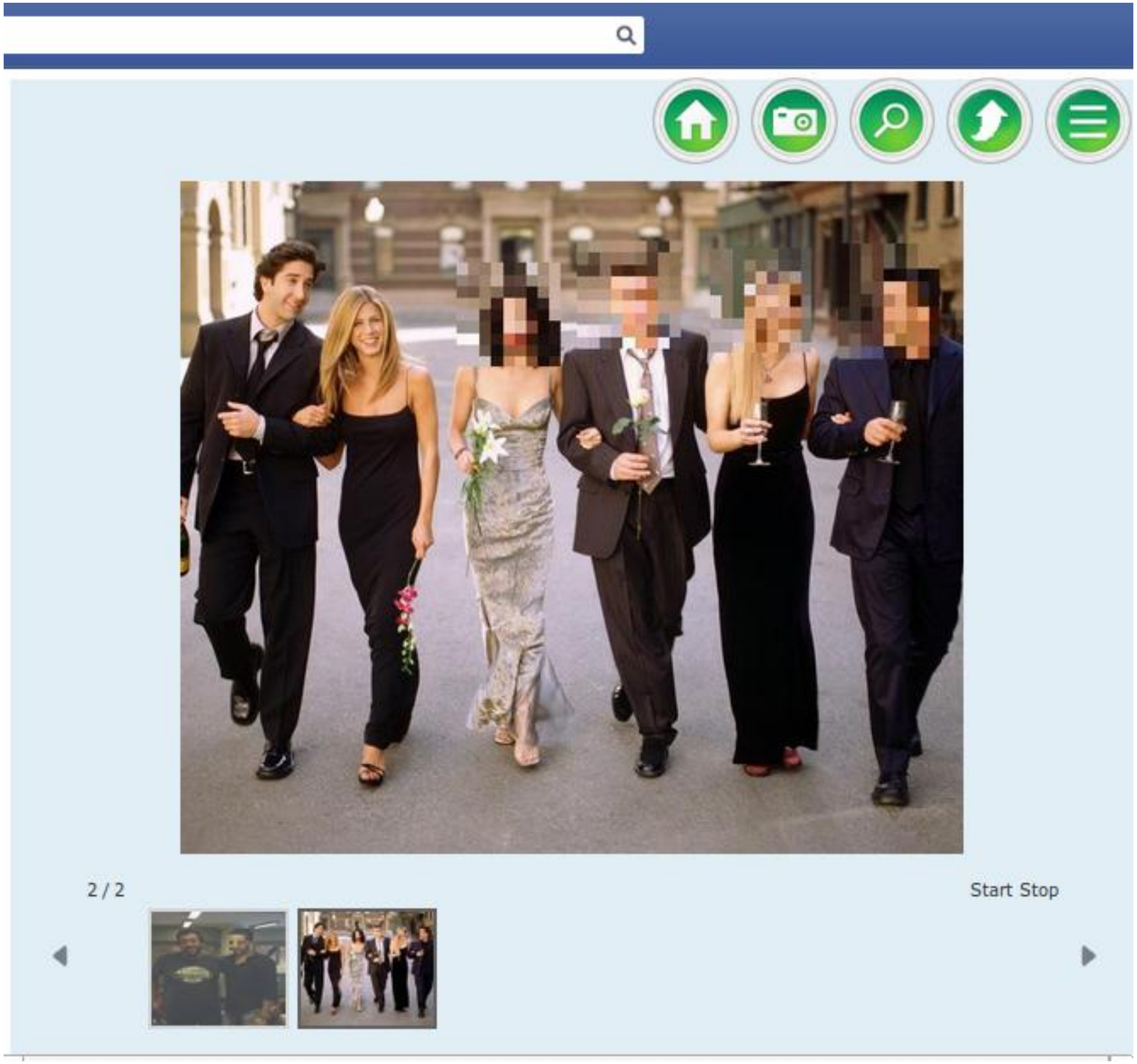
We propose a new fine-grained access control mechanism.

- Enforce **face-level** access control (according to user's access-list).
- Handles effectively the **conflicting visibility settings** of the users.
- Can **inter-operate** with the existing access control mechanisms.

We demonstrate its applicability with a Proof-of-concept application.

iSocial

Preserving user privacy in shared content



FORTH
Institute of Computer Science